

Floquet Color Code について

[2] に示されている、Floquet Color Code についてどのような性質があるのか調べてみた。

1. Color Code

Color Code(CC) は Fig.1 に示されるようなもので、六角形を平面上にならべ、隣り合う六角形の面は異なる色で塗ったものである。量子ビットは六角形格子の点の上に配置する。スタビライザー s はそれぞれの色の面に対応していて、色を c 、面を p とすると、以下の式で表される。

$$s_{c,p}^x = \prod_{j \in \text{vert}(p)} X_j, \quad s_{c,p}^z = \prod_{j \in \text{vert}(p)} Z_j, \quad (1)$$

ここで、 $\text{vert}(p)$ は p 面に属する点（角）に対応する。Fig.1 の例だと、量子ビットが 37 個、スタビライザーが 36 個で、1 量子ビットをエンコードできることがわかる。

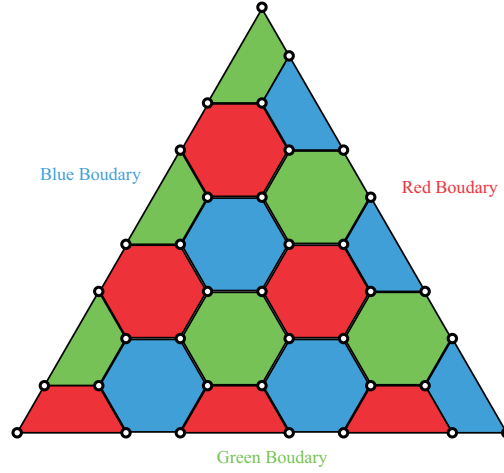


Fig. 1

また、これらのスタビライザーをハミルトニアンとして、

$$H_{CC} = - \sum_{c,p} s_{c,p}^x - \sum_{c,p} s_{c,p}^z \quad (2)$$

というふうには書くとエラーが生じている状態を excitation として、anyon 粒子を構成する事ができる。赤、緑、青をそれぞれ r,g,b と表すと、c 面の x スタビライザーが exite したときは cx という粒子、c 面の z スタビライザーが exite したときは cz という粒子、c 面の x と z スタビライザーが exite したときは cy という粒子というふうに（c には r,g,b のどれかが入る）、9 つの anyon 粒子を定義することができ、それを表にして Tab.1 に示す。

	r	g	b
x	rx	gx	bx
y	ry	gy	by
z	rz	gz	bz

Tab. 1

Fig.1 の boudary はこの anyon 粒子のうち、どれが condense されるかに基づいている。ただし、説明が長くなるので具体的な性質については [2][3] を参照。

このような anyon model は Toric Code(TC) にも適用でき、X スタビライザーの excitation を e 粒子、Z スタビライザーの excitation を m 粒子として表す。また、e は smooth boudary で condense され、m は rough boundary で condense される。

2. Floquet Color Code

Floquet Color Code は Fig.1 の CC で、一つの色と Pauli を選んで、Fig.2 のようにスタビライザーを選んだものである。ここで、赤線は 2-weight の X スタビライザー、赤字 X の面は 6-weight の X スタビライザー、青と緑の面は 6-weight の Z スタビライザーである。境界のスタビライザーは、扇形に属する点への X または Z スタビライザーである (おそらく)。また、境界にある赤点は 2-weight の X スタビライザー (赤線) が 1-weight になったものである。おそらくこの例では、符号距離が 8 である。

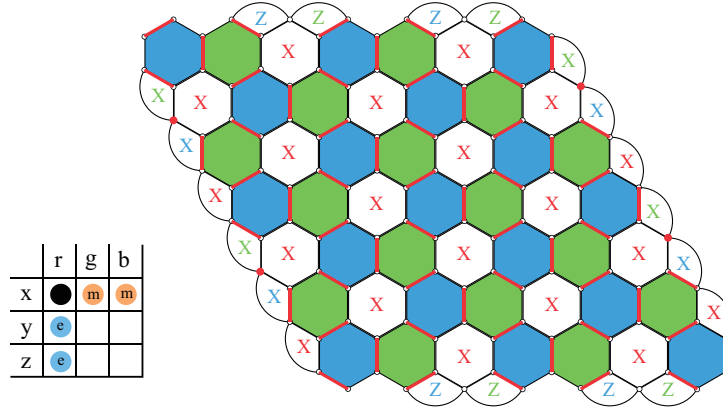


Fig. 2

Fig.2 の例が [2] に示されているが、量子ビット数が 126 個に対して、スタビライザーが 132 個あるため矛盾 (私がスタビライザーの定義を読み間違えている可能性がある)。[2] の論文内では、赤面を Z スタビライザーにしたり、他の色でも同じように定義できることが示されている。また、変え方に順番はあるが、Floquet Code として使う色を変えたとしても logical operator が保存されることが示されている。

Fig.2 の場合だとスタビライザーが多すぎるので、違う例を考えてみた。rotated surface code のような感じで、Fig.2 の例の無駄な部分 (符号距離に関係ない部分) を省いて、いい感じにスタビライザーを追加すると Fig.3 のようになった。いずれにしても、赤が X スタビライザーで緑と青が Z スタビライザーである。Fig.3 では、量子ビットが 67 個、スタビライザーが 66 個なので、1 量子ビットをエンコードできるようになっている。また、Fig.2 の例についても同様だが、Fig.3 では CC の anyon 粒子のうち、ry,rz,gx,bx だけを残したモデルになっており、ry,rz が TC の e 粒子、gx,bx が TC の m 粒子と対応している。すなわち、このような Floquet Code は TC である。そして、Fig.3 で e が Rough Boundary で condense、Smooth Boundary で confine され、m が Smooth Boundary で condense、Rough Boundary で confine されることがわかる。また、平行四辺形の 1 辺が符号距離に対応していて、Fig.3 の場合は符号距離が 7 である。

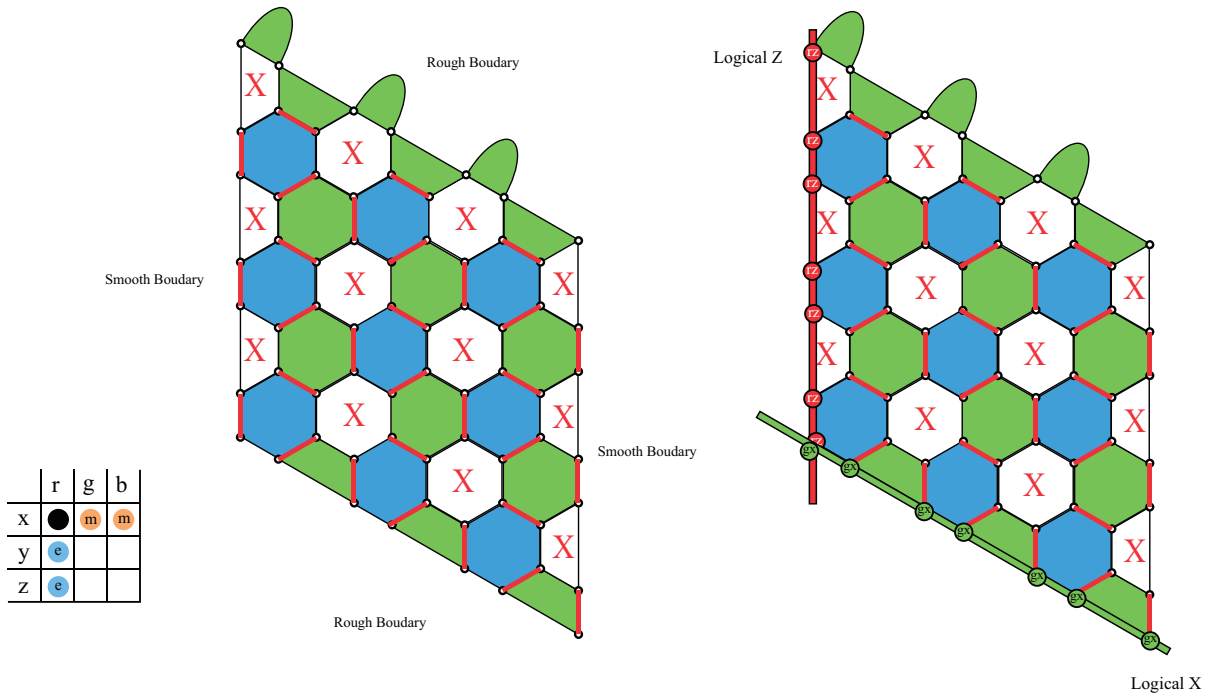


Fig. 3

Fig.4 のように緑の Z を選ぶ場合も同じグラフで実現することができる。Fig.3 と Fig.4 では anyon 粒子を共有している部分があるため Logical Operator を保存している。実際にグラフから確かめても、Logical Operator は保存されている。赤 X から青 Z への変換も考えたが、うまく定義できなかった。

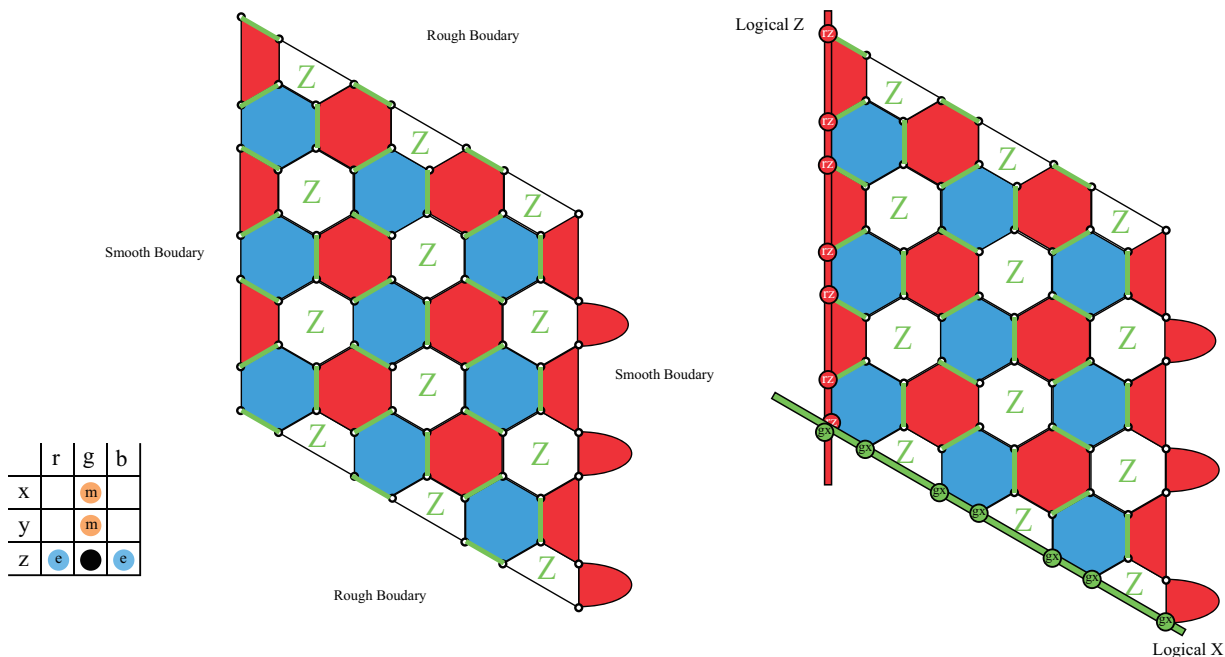


Fig. 4

3. Unfolding Color Code

[4] に CC を unitary transformation すると、2 つの重なった TC で表すことができることが示されている。このように実現される CC は TC が折りたたまれたような構造になっている。[1] でも指摘されているが、[4] の unitary transformation では符号距離が保存されない。また、そもそも計算の途中で unitary transformation がどのように実行されるのかを私は理解していない。[1] の escape stage では、やむを得ず CC と TC の lattice surgery を行って、符号距離を大きくしている。そこで、効率がいいのか全くわからないが、Floquet Color Code の一つの例である Fig.3 の符号を 2 つ使って CC を実現し、Magic State Cultivation を行えると思う。最初から、2 つの TC で CC を実現すれば、escape stage の grafting が格段に楽になると思った。具体的には、以下のような TC を用意する (Fig.5)。

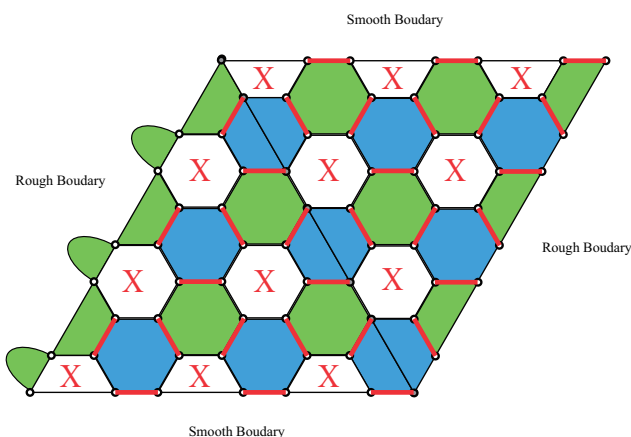


Fig. 5

Fig.5 では平行四辺形の短い対角線が示されているがこれが折り目になっている。よって、CC を実現させるときはこの折り目で折りたたみ、Fig.6 を作る。

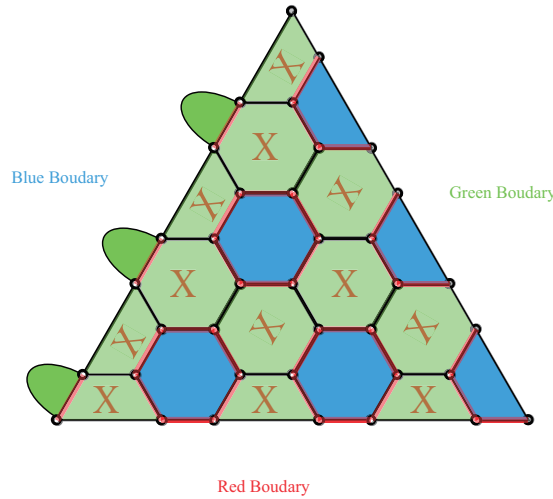


Fig. 6

また、このとき TC の anyon 粒子を組み合わせることで、CC の anyon model を再現することができる。その組み合わせを Tab.2 に示す。ただし、添字の + は折りたたんだときの上の TC、- は折りたたんだときの下の TC を表す。また、 $f = e \times m$ である。

	r	g	b
x	e^-	e^-e^+	e^+
y	e^-m^+	f^-f^+	m^-e^+
z	m^+	m^-m^+	m^-

Tab. 2

たしかにこのようにすることによって、condense や confine が CC と同じように起こることがわかる。そして、Lattice Surgery は TC の Lattice Surgery と同じようにして行うことがおそらくできる。

4. まとめ

今回示した Floquet Code を折りたたむような方法を用いれば、Magic State Cultivation を効率よく行えるかもしれない。今回は 3 色で Floquet Code を実装できていない (赤 X から青 Z への変換が思いつかない) ので [2] に示されている完全な Floquet Color Code を実現できていない。Floquet Code について大して勉強したことがないので、途中で符号を変えることの利点をまだ理解していないが、それらを調査して、このような折りたたんだ Floquet Color Code に優位性があるのかないのかを確かめていきたい。

REFERENCES

- [1] Craig Gidney, Noah Shutty, and Cody Jones. “Magic state cultivation: growing T states as cheap as CNOT gates”, arXiv preprint. eprint: 2409.17595. URL: <https://arxiv.org/abs/2409.17595>.
- [2] Markus S. Kesselring, Julio C. Magdalena de la Fuente, Felix Thomsen, Jens Eisert, Stephen D. Bartlett, and Benjamin J. Brown. “Anyon Condensation and the Color Code”, PRX Quantum **5**, 010342. DOI: 10.1103/PRXQuantum.5.010342. URL: <https://doi.org/10.1103/PRXQuantum.5.010342>.
- [3] Markus S. Kesselring, Fernando Pastawski, Jens Eisert, and Benjamin J. Brown. “The boundaries and twist defects of the color code and their applications to topological quantum computation”, Quantum **3**, 128. URL: <https://quantum-journal.org/papers/q-2018-10-19-101/>.
- [4] Aleksander Kubica, Beni Yoshida, and Fernando Pastawski. “Unfolding the color code”, arXiv preprint. eprint: 1503.02065. URL: <https://arxiv.org/abs/1503.02065>.